

3D ábrázolás – PoVRay (2. rész)

3D világunk fontos része a fény, s ehhez szükség van legalább egy fényforrásra, hogy lássuk is a térben elhelyezett tárgyakat. Fényforrást illetően több lehetőség között választhatunk, amelyek közül néhányat megtalálunk a valódi világban is. Természetesen léteznek olyan fényforrások is, amelyek csak a számítógép képi világában léteznek: ezekkel speciális effekteket tudunk létrehozni.

A valódi világban minden fényforrás létező tárgy is egyben, amely olyan jellegű fényt bocsát ki, ami rá jellemző: ilyen az izzólámpák széles spektrumú – pontszerű, a fénycsövek monokromatikus szórt jellegű, illetve a hidegtükrös halogén izzók erős fénykúpja és kismértékű fényudvara. A 3D világban a fényforrásoknak vannak ilyen jellemzőik, azonban nem rendelkeznek tárgyas tulajdonságokkal: nem léteznek, csak a fényük árulja el őket. Ennek egyik mellékhatása, hogy nem okoznak megcsillanást, illetve nem vakítja el a kamerát a fényük. Ez többnyire előnyös, néha azonban a valósághoz hasonlóan előnyös lenne, ha a kamera úgy viselkedne, mint egy kamera, vagy a fényforrás lenne valóságosabb.

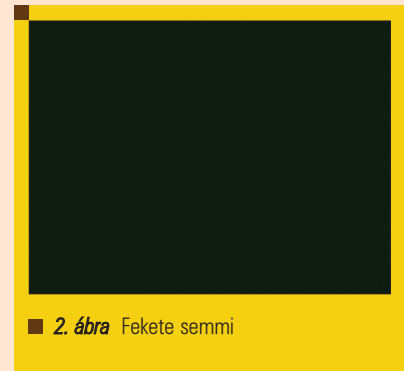
A 3D helyszínen fényforrásoknak két alapvető tulajdonságuk van: a térben elfoglalt pozíciójuk, és a színük. A legegyszerűbb fényforrással már találkoztunk is:

```
light_source{
  <0,0,0> color white}
```

Ez a fény a koordináta rendszer origójában található és fehér színű fényt bocsát ki magából. Ha egy kamerát is hozzácsapunk az *SDL* állományhoz, akkor megvizsgálhatjuk a fényforrás és kamera mellékhatását is (*pov07.pov*). A kapott képen ugyanis tömör feketeséget látunk csupán, holott a fényforrásunk teljes erővel „világít”. Ha elhelyezünk egy tetsző-



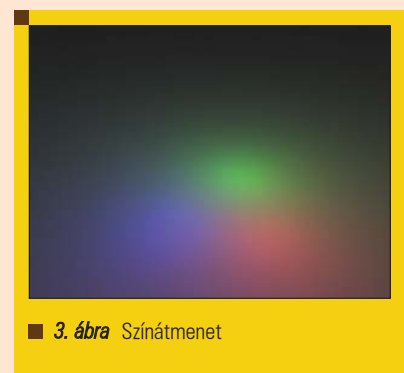
■ 1. ábra Zöld síkon fehér fényforrás



■ 2. ábra Fekete semmi

leges testet, amelyet a fény megvilágíthat, s onnan visszaverődhet, akkor már látni is fogunk a képen valamit (feltéve, ha nem a tárgy belsejébe tettük a fényforrást (*pov08.pov*, 1. ábra). Nem meglepő módon a fehér fény zöld pacaként látszik, hiszen a létrehozott sík zöld színű, így a fehér fényben zöld színűnek *kell* látszania. Látható a sík megvilágításából, hogy a fényforrás felette található, viszont a kamerába nem csillan bele a fény. Ezen a tényen jelenleg még nem változtatunk, fogadjuk el ezt a működésmódot.

Ha a fehér fény helyett vörös színű a fényforrásunk (*pov09.pov*, 2. ábra), akkor érdekes módon a zöld tárgyak fekete színűek lesznek. Ennek oka, hogy a tükröződésmentes tárgyak (amelyeket alapesetben a *PoVRay* készít) a fényt elnyelik, majd a saját színkomponensüknek megfelelő részét kibocsátják: a vörös fényt elnyeli a zöld színű síkunk, majd a kapott fény zöld összetevőjét bocsátja ki. A vörös fény



■ 3. ábra Színátmenet

azonban nem tartalmaz zöld komponenset, így kaptunk tömör sötétséget eredményül.

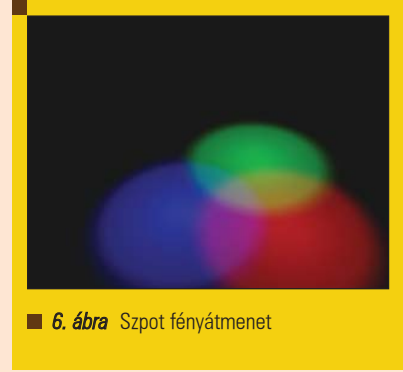
Érdekes játék a színekkel (3 ábra, *pov10.pov*), ha egy fehér síkot a három alapszínnel világítunk meg, ekkor ugyanis a színek találkozásánál az összes lehetséges *RGB* (*Red-Green-Blue*) színrendszerbeli kombináció előkerül, ha elegendően nagy felbontással dolgozunk (24 bites színmélység esetén minimum 16.7 millió pixel).



■ 4. ábra Szpot fényforrás



■ 5. ábra Éles szélű szpot fényforrás



■ 6. ábra Szpot fényátmenet



■ 7. ábra Egy egységnyi magasan



■ 8. ábra Két egységnyi magasan

```
1. lista – pov14.pov

light_source{
<1,1,0> color Red
cylinder point_at <1,0,0>}

light_source{
<-1,1,0> color Red
spotlight point_at <-1,0,0>}
```

A pontszerű fényforrás

A *PovRay* alapesetben pontszerű fényforrást alkalmaz, ha elhelyezünk egyet a térben. Ez a tér minden pontja felé azt a színű fényt sugározza, amelyet meghatároztunk a számára. Ennek a típusnak nincs semmi különös beállítási lehetősége, a közös fényforrás módosítások érvényesek rá, amelyeket később részletezek. A fenti példákban ilyen pontszerű fényforrást alkalmaztunk.

A szpot fényforrás

A hidegtükrös halogénizzók mintájára beállíthatunk olyan fényforrást, amely mindössze egy meghatározott szögben sugároz fényt. Nézőpont hiányában teljesen biztos, hogy nem a meg-

```
2. lista – pov15.pov

light_source{
<1,2,0> color Red
cylinder point_at <1,0,0>}

light_source{
<-1,2,0> color Red
spotlight point_at <-1,0,0>}
```

felelő irányba fog fényt sugározni, így mindenképpen adjunk meg egyet a *point_at* kulcsszóval (*pov11.pov*, 4. ábra).

```
light_source{
<0,1,0> color Red
spotlight point_at <0,0,0>}
```

Ha nem határozunk meg egyéb értéket, akkor a szpot fényforrás 30 fokos szögben bocsát ki fénysugarakat, amelyhez még 15 foknyi fényátmenet tartozik. A fénykör paramétereit a *radius*, a *falloff* és a *tightness* kulcsszavakkal lehet beállítani. A *radius* adja meg közvetve a fénykör sugarát, mivel a mértékegysége fok, s ez a szög határozza meg a kibocsátott

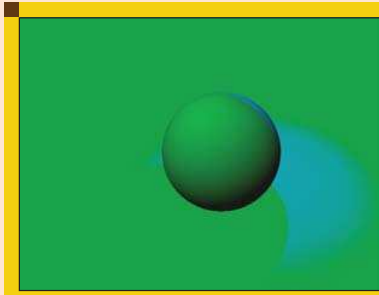
fénykúp nyílásszögét. Ha a *falloff* szöget nagyobbra vesszük, mint a *radius* értéke, akkor a kibocsátott fény a két szög által meghatározott tartományban teljesen elenyészik, így egy átmenet látszik. Ha a *falloff* értéke kisebb vagy egyenlő a *radius* értékével, akkor a fénykör éles peremmel szűnik meg, nem lesz szórt fényű udvara (*pov12.pov*, 5. ábra).

```
light_source{
<0,1,0> color Red
spotlight point_at <0,0,0>
falloff 25}
```

Ha szpot fényforrással készítjük el a három színű sugárzó képünket (*pov13.pov*, 6. ábra), akkor a képen találunk olyan területeket, ahol a három szín azonos intenzitással fedi a neki szánt területet, és a találkozásoknál a szín csak a kettő (vagy három) szín tiszta találkozásából áll átmenet nélkül (zöld-vörös, zöld-kék, kék-vörös színfolt). A *falloff* által meghatározott szögön kívül már nem látunk fényt, a két szög között pedig láthatjuk a színek keveredését. Ha nem szeretnénk, hogy a szpot területen a lefedettség egyenletes legyen, akkor a *tightness* értékének (alapértéke 0, s a maximális értéke 100 lehet) megváltoztatásával, már a fénykúpon belül is csökken a fény intenzitása a szélek felé haladva.

Hengeres fényforrás

A hengeres fényforrás szinte teljesen azonos a szpot fényforrással, azonban a fénykör átmérője nem függ a fényforrás távolságától. A *pov14.pov* részlete (7. ábra) szerint a fényforrások 1 egységnyi magasan vannak a síktól, míg a *pov15.pov* részlete (8. ábra) szerint már két egységnyire.



■ 9. ábra Arnyékvetés



■ 10. ábra Az árnyék kikapcsolva



■ 11. ábra Tárgyon át vetített fény

Látható, hogy a szpotfény átmérője megnövekedett, míg a hengeres fényforrás fényköre nem. Ezekkel a fényforrásokkal a lézertényhez hasonló fényeket tudunk készíteni, amelyek nagy távolságokon át is megtartják az átmérőjüket. Fontos különbség azonban, hogy a fénysugarak nem párhuzamosak, hanem a megadott pontból indulnak, csak nem tudnak kilépni a paraméterekkel meghatározott hengerből (ez az átlátszó tárgyak fénytörésénél okozhat problémát).

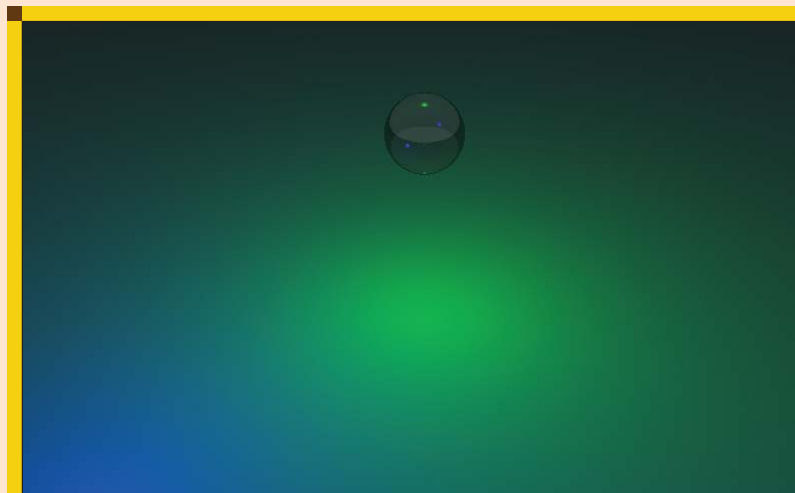
Párhuzamos fényforrás

Mind a három fényforrás fénye párhuzamosra tehető, amely akkor tesz jó szolgálatot, ha ilyen jellegű fényre van szükségünk; a napfény például gyakorlatilag párhuzamos fénysugarakból áll. Bármelyik fényforrással készíthetünk szinte párhuzamos fénysugarakat, ha nagyon messze tesszük a tárgyainktól (mint amilyen távol a Nap található a Földtől). Ezen megoldással mindössze annyi a gond, hogy a távoli fényforrás fénye gyér, meg kell sokszorozni, hogy hatása legyen. Egyszerűbb hozzáfűzni a fényforrás leírójához a *parallel* kulcsszót, és a fényforrás a megadott pont közelségének intenzitásával fog párhuzamos nyalábokat sugározni (*pov16.pov*).

```
light_source{
  <-1,0,0> color Green
  parallel point_at <-1,-1,0>}
```

Árnyékok

A sugárkövetés tálcán kínálja az árnyékok ábrázolását, mivel a tárgyak eltakarják a fény útját, ahol ebből következően árnyék keletkezik (*pov17.pov*, 9. ábra). Az árnyék éles körvonallal rendelkezik, akkor is, ha a fényforrás elhalványuló része vet



■ 12. ábra Tárgyba épített fényforrás

árnyékat. Egy kicsit természetellenesen hat, mivel a valóságban az elhalványuló fénykör a fényforrás nem pontszerű jellege miatt keletkezik, azonban ez túl sok számítási kapacitást igényel (lehet javítani ezen, de erről pár bekezdéssel később foglalkozunk majd).

Ha egy tárgy leírójában kikapcsoljuk az árnyékokat, akkor a fény keresztül tud haladni rajta mindenféle fénytörés nélkül, ugyanakkor a felülete el is nyeli azt (érdekes probléma az energiamegmaradás terén :) (*pov19.pov*, 10. ábra).

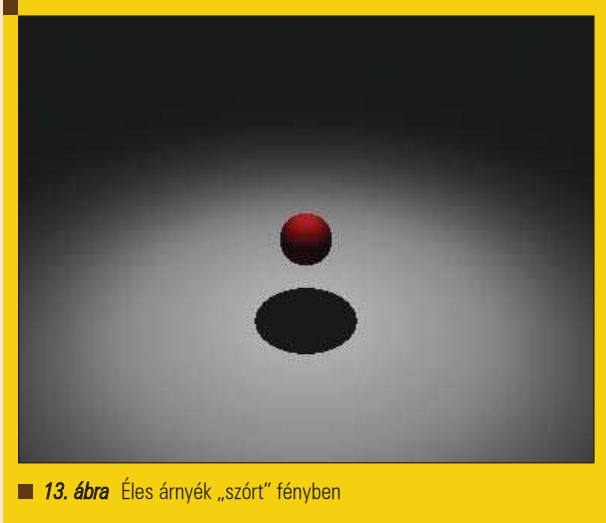
```
sphere{
  <0,0,0>,1
  texture{pigment{color white}}
  no_shadow}
```

Ugyan *nem* az árnyékok kezelése módosítható a fényforrás leírójába írt *shadowless* kulcsszó használatával, a program készítői ezt a nevet adták ennek a kulcsszónak. Ebben

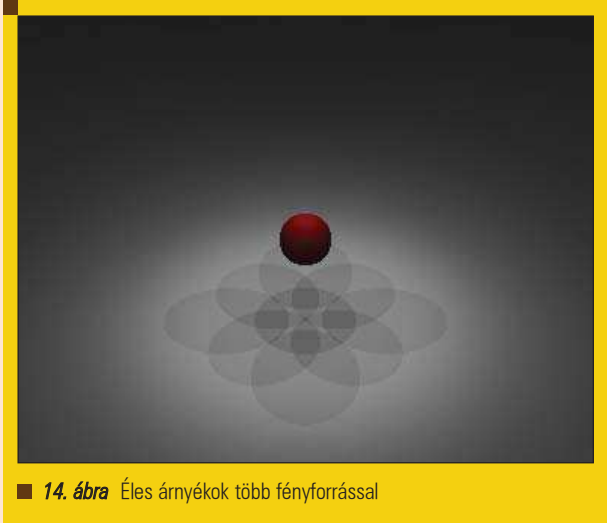
az esetben a megadott fény szórt háttérfénnyé válik (*pov18.pov*), s ezzel egyfajta kellemes háttérfényt tudunk varázsolni különösebb problémák nélkül.

Alakzat vetítése

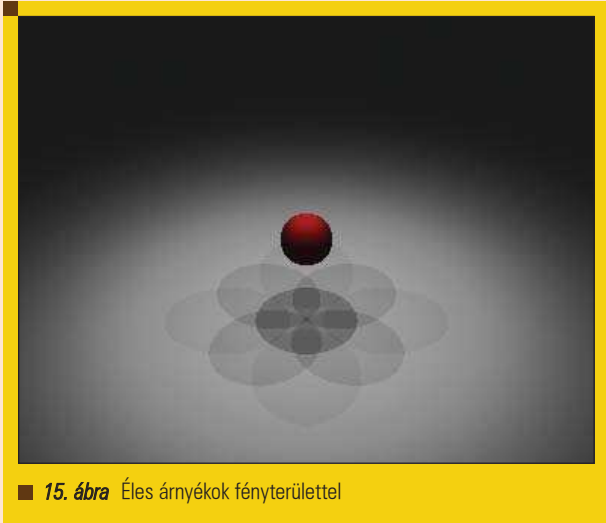
Az alakzat vetítését nevezhetjük akár negatív árnyéknak is, ugyanis a fény csak arra halad tovább, ahol az adott testen át tud menni. A megadott testnek nem kell átlátszónak lennie ahhoz, hogy a vetítés működjön. Egyszerűen csak a fényforrás leírójába kell helyezni a *projected_through* kulcsszó után azt a testet, amelyen át szeretnénk a fényt vetíteni. Ezzel a megoldással olyan összetett fényforrásokat tudunk készíteni, amelyekre csak szükségünk lehet (érdekes megoldás például ködbe szöveget vetíteni :). Ha két gömböt összevonunk, majd ezen át vetítünk a szpotfényt (*pov20.pov*), akkor a (11. ábra) szpotfény egy része élesen ér véget, a többi része pedig a fény elhalványulásával.



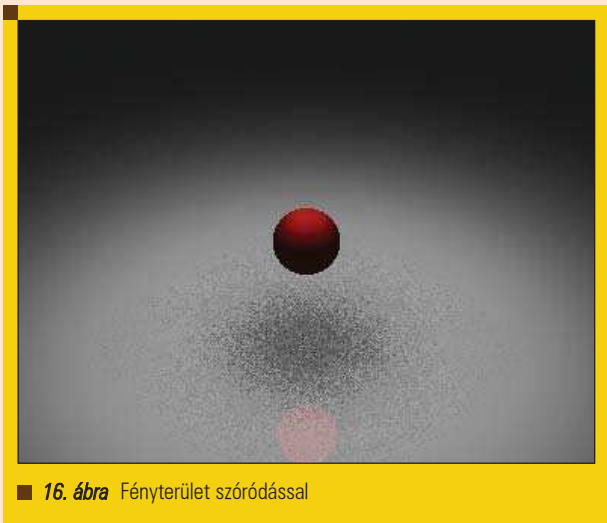
■ 13. ábra Éles árnyék „szórt” fényben



■ 14. ábra Éles árnyékok több fényforrással



■ 15. ábra Éles árnyékok fényterülettel



■ 16. ábra Fényterület szóródással

```
light_source{
  <0,2,0> color Green
  spotlight
```

```
projected_through{union{sphere{
  ↪ <-0.4,1,0>,0.5}
  ↪ sphere{<0.4,1,0>,0.5}}}}
```

Fényforrás építése

Készíthetünk olyan fényforrást, amely látható lesz számunkra, és a fény belőle árad majd. Ehhez egyszerűen egy testet kell készítenünk, amely villanykörteként fog funkcionálni, vagyis tartalmazni fogja a fényforrást. Ha ezt a testet hozzáadjuk a kiszemelt fényforráshoz a `looks_like` kulcsszó után, akkor a fény szabadon átjut rajta. Ha a test tömör vagy átlátszatlan, csak ekkor más fény hiányában tömör fekete lesz az elkészült képen, ezért lehetőleg üvegből készítsük el (*pov21.pov*, 12. ábra).

3. lista – pov24.pov

```
light_source {
  <0, 1, 0> color white
  area_light <1, 0, 0>, <0, 0,
  ↪ 1>, 3, 3

  spotlight point_at <0, 0, 0>
  tightness 0
  radius 40
  falloff 70}
```

```
light_source{
  <0,1,0> color Green
  looks_like{sphere{<0,1,0>,0.5
  ↪ texture{T_Glass1}}}}
```

Ezzel a módszerrel sem tudjuk igazából „láthatóvá” tenni a fény forrását, hiszen

4. lista – pov25.pov

```
light_source {
  <0, 1, 0> color white
  area_light <1, 0, 0>, <0, 0,
  ↪ 1>, 3, 3
  adaptive 5
  jitter

  spotlight point_at <0, 0, 0>
  tightness 0
  radius 40
  falloff 70}
```

nem vakítja el a kamerát, és „zavaró” becillanások sincsenek a képen: mindössze a megadott testből árad a fény, ahogy a villanykörteből szokott.



■ 17. ábra Globális fények kikapcsolva

Fényterületek

Ha igazi elmosott árnyékot akarunk kapni (tehát nem olyan esetet, mint a *(pov22.pov, 13. ábrán)*, akkor pontszerű fényforrások helyett nagyobb kiterjedésű fényforrást kellene használnunk, ugyanis a valódi világban minden fényforrásnak van valamekkora kiterjedése. Minél nagyobb egy fényforrás, annál életokebbek az árnyékok, ezáltal a szemnek kellemesebb képeket tudunk készíteni. A sugárkövetés működés módja miatt egy fényforrással többet tartalmazó világ egy egységnyi idővel több ideig tartó renderelési időt is jelent. Ebből következően a nagyobb kiterjedésű fényforrásokat pontszerű fényforrások egymás mellé helyezésével tudjuk „modellezni”.

Ha a fényforrások számát megnövekedtetjük és elhelyezkedését megfelelően módosítjuk (*pov23.pov, 14. ábra*), akkor kicsit előbbre tudunk jutni, viszont a kapott eredmény szerint úgy járunk, mint a focisták a stadionban: négy éles árnyékuk lesz (a képen látható gömbnek pedig kilenc).

```
light_source {
  <-0.5, 1, -0.5> color
  ↳ <0.11,0.11,0.11>
  spotlight point_at <0, 0, 0>
  tightness 0
  radius 40
  falloff 70}
```

[...]

Mellékhatásként a fényforrások intenzitását annyiadrészére kell csökkentenünk, ahány fényforrást alkalmazunk, különben a kép túlvilágított lesz. Tehát ez a megoldás sem igazi,

5. lista – pov26.pov

```
light_group{
  light_source {
    <0, 2, 2> color white
    spotlight point_at <0,0,0>}

  sphere{
    <0,1,1>,0.5
    texture{pigment{color Red}}}
```

global_lights off}

sok munkát jelent, és mégsem ad megfelelő minőséget: sokkal finomabban kellene a fényeket elhelyezni, amely sokkal több számítási kapacitást igényel a kép számolása közben. A *PovRay* képes saját hatáskörén belül megoldani, s erre az *area_light* kulcsszóval tudjuk utasítani. Ezzel létrehoz egy kétdimenziós tömböt fényforrásokból, amelyet használva gyakorlatilag azonos képet állítva elő (*pov24.pov, 15. ábra*).

Az előnye viszont az, hogy ezt egy sorban el tudjuk intézni, nem kell a fényforrások definícióját többször megismételni, amely a módosítást erősen megnehezíti. A *PovRay* saját fényterületének előnye, hogy képes egy kicsit összekavarni a „fényugarakat”, és így tényleg olyan lehet az árnyék, mintha nagyobb kiterjedésű fényforrás világítana (*pov25.pov, 16. ábra*). Ehhez mindössze két kulcsszót kell beleírunk a fényforrás leírójába: ez a *jitter* és az *adaptive*. A *jitter* kell ahhoz, hogy az árnyék tényleg szórt legyen, az *adaptive* pedig a megadott szám hatványa szerint további fényforrásokat készít, ha szükséges.

Megvilágítási csoport

Ritka eset, de néha szükséges olyan megvilágítás, ahol egy-egy tárgyat világít meg egy fényforrás. Ebben az esetben egy csoportba gyűjtjük a fényforrásokat és azokat a tárgyakat, amelyeket ezek a fények fognak megvilágítani. Ezt a gyűjtést a *light_group* kulcsszó vezeti be. Ennek eredménye, hogy *csak* a csoportba foglalt testeket világítják meg azok a fényforrások, amelyek

a csoportban szerepelnek. A többi testen nem lesz hatása ezeknek a fényforrásoknak, ellenben a globális fények hatással vannak a csoportba rendelt testekre. Ez utóbbit kikapcsolhatjuk a *global_lights off* parancs segítségével (*pov26.pov, 17. ábra*).

Fontosabb paraméterek

A *PovRay* program használatához érdemes a parancssoros program néhány paraméterét ismerni, mivel így jelentős sebességnövekedést vagy minőségi ugrást tudunk elérni. Alap esetben a gyorsaságra van beállítva a program, tehát vannak olyan részek, ahol a kapott eredmény eléggé szegényesen néz ki.

Érdemes beállítani, hogy egy rettentő pixeles, de átlátható képet kapjunk a renderelés folyamatáról, így a durva hibákat azonnal ki tudjuk szűrni, és nem kell megvárni, amíg több perc munka után észrevesszük, hogy hiányzik egy test a képről. Ehhez a *+SPn* és a *+EPn* paramétereket kell megadni, ahol a *+SP* a első lépés pixelben való méretét kéri számszerűen, a *+EP* pedig az utolsó lépést. Érdemes 32 pixellel kezdeni és 4-8 pixel körül végezni ezzel a finomítással, mert 2-4 pixel körül már eléggé részletesen látszik minden szükséges test.

A kész kép minőségét a *+Qn* paraméterrel tudjuk befolyásolni, ahol a szükséges szám 0 és 9 között lehet közönséges sugárkövetésnél, és 10 vagy 11 lehet a speciális sugárkövetést választva.

A következő rész témája a primitív testek és a testek közötti kölcsönhatás lesz, ahol a kölcsönhatás alatt a különféle „halmazműveleteket” értjük.



Auth Gábor

(auth.gabor@enaplo.hu)

Egy pécsi középiskolában informatikát és programozást oktat.

Tíz éve botlott először

a UNIX rendszerekbe, 7 év Linux használat után kapta el a FreeBSD lázat, amiből máig nem tudott kigyógyulni.

KAPCSOLÓDÓ CÍMEK

A PovRay projekt honlapja

↳ <http://www.povray.org>